

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant: Steven M. Trimberger et al.
Assignee: Xilinx, Inc.
Title: "Partially Encrypted Bitstream Method"
Ser. No.: 09/724,974 Filing Date: 11/28/2000
Examiner: Nguyen Art Unit: 2137
Docket No.: X-805-3-US Conf. No.: 7823

COMMISSIONER FOR PATENTS
P.O Box 1450
Alexandria, VA 22313-1450

DECLARATION UNDER 37 C.F.R. §1.131

Dear Sir:

I hereby state and declare that I, Walter N. Sze, am a joint inventor of the subject matter which is claimed and for which a patent is sought on the invention entitled, "Partially Encrypted Bitstream Method", having Application Serial Number 09/724,974, and filed on November 28, 2000.

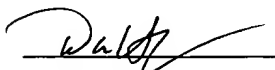
I, Walter N. Sze, further state and declare that I have reviewed and understand the contents of the above-identified specification, including the claims, and that:

1. The invention claimed in the above-referenced application was conceived before December 22, 1999. Attached is a true and accurate copy of an electronic mail message that was drafted and sent prior to December 22, 1999 and that describes the claimed invention, with the date and unnecessary names and email addresses redacted.

2. From before December 22, 1999, until successful testing and implementation, I was employed and personally involved with other engineering employees of Xilinx, Inc., in an ongoing engineering effort to successfully design and construct a field programmable gate array (FPGA) known as the Virtex II FPGA. The Virtex II FPGA was constructed and operated according to the invention claimed in the above-referenced patent application, Serial No. 09/724,974.

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Signed this 19 day of August, 2004, by:



Walter N. Sze

Witnessed this 19th day of August, 2004, by:

Signature: 

Julie A. Matthews

Subject: Re: V2 Bitstream encryption

Date: [REDACTED]

From: Raymond Pang [REDACTED]

To: [REDACTED]

CC: [REDACTED]

Hi [REDACTED],

The encryption algorithm is a "multi-pass/multi-key DES using an initial CBC value in outer CBC mode." Steve Trimberger can point you to a book which describes this in detail.

Only the design portion of the bitstream (i.e., that portion going to the FDRI register) should be encrypted. We've created a new OP code, DECRYPT, to be used instead of the WRITE OP code for encrypted data. When the Packet Processor sees DECRYPT in a header, it will know that ensuing data is encrypted and will redirect that data to the decryptor before sending it to the FDRI register.

But even before encrypted data is sent, the decryptor needs to be prepared by telling it the initial key address, number of passes, and initial CBC value, and then initializing it, all through a series of configuration register writes. Walter Sze has information about all the V2 config registers on the V2 web page (<http://web/randd/hardware/blue>). As to how the user enters all this information to the software, I have no idea.

Because the algorithm works on 64 bit chunks, encrypted data needs to be aligned to 64 bit boundaries and because config data is in 32-bit words, the first 32 bits need to be on the lower portion of this boundary. For reasons having to do with cores, IP's, and partial reconfig. capabilities, we like to limit the encrypted data granularity to a frame (which in V2 happens to be a multiple of 64 bits.)

Decryption will be supported for all config modes.

That's all I can think of for now. Does anyone have anything to add or correct? Let me know when you have more questions.

-Raymond

[REDACTED] wrote:

> Hi Raymond,
> I am trying to estimate the amount of work involved to add encryption to the V2
> bitstreams. Do you know the encryption algorithm will be and how the bitstream
> data will be processed? Is there something else that bitgen will need to do besides
> encrypt the data (i.e. set bits in the control register, limit the configuration freq.).
> We are just trying to get an idea of what is involved so if you have any general
descriptions
> of the encryption, that would be great.
>
> [REDACTED]